

Un Modelo ACO para una Versión No Estacionaria del Problema del Ascensor Único

S. Molina, G. Leguizamón

Universidad Nacional de San Luis,
Ejército de los Andes 950, (5700) San Luis, Argentina
{smolina,legui}@unsl.edu.ar

y

Enrique Alba

Universidad de Málaga
Complejo Tecnológico - Campus de Teatinos, 29071
Málaga, España
eat@lcc.uma.es

Abstract

The Ant Colony Optimization (ACO) metaheuristic is a bio-inspired approach for hard combinatorial optimization problems for stationary and non-stationary environments. In the ACO metaheuristic, a colony of artificial ants cooperate for finding high quality solutions in a reasonable time. An interesting example of a non-stationary combinatorial optimization problem is the Multiple Elevators Problem (MEP) which consists in finding a sequence of movements for each elevator to perform in a building so that to minimize, for instance, the waiting average time of the passengers. Events like the arrival of one new passenger to the elevator queue or the fault of one elevator produce dynamically the changes of state in this problem. A subclass of MEP is the non-stationary version of the so called Single Elevator Problem (SEP). In this work, we propose the design of an ACO model for the SEP that can be implemented as an Ant Colony System (ACS).

Keywords: Ant Colony Optimization, Single Elevator Problem (non-stationary version), Ant Colony System design.

Resumen

La técnica Optimización de Colonias de Hormigas (ACO) puede ser aplicada a problemas de optimización combinatorios duros en ambientes estacionarios y no estacionarios. En dicha metaheurística, las hormigas artificiales de una colonia cooperan para encontrar soluciones de alta calidad en un tiempo razonable. Un ejemplo interesante de un problema de optimización combinatorio no estacionario para ser abordado a través de una técnica ACO, es el *Problema de los Ascensores Múltiples* (MEP) el cual consiste en encontrar la secuencia de movimientos que debe realizar cada ascensor de un edificio de manera tal de minimizar el tiempo medio de espera de los pasajeros. La llegada de un nuevo pasajero a la cola de un ascensor, la rotura de un ascensor, etc, son los eventos que provocan cambios de estado en este problema, haciéndolo dinámico. Una subclase del problema de ascensores múltiples es el denominado *Problema del Ascensor Único* (SEP) en su versión no estacionaria. En dicho problema se considera la existencia de un único ascensor en un edificio en donde no se tienen en cuenta las ocurrencias de eventos. En este trabajo se presenta un modelo ACO para SEP y en base a éste modelo, se propone el diseño de un *Sistema de Colonias de Hormigas* (ACS).

Palabras claves: Optimización de Colonias de Hormigas, Problema del Ascensor Único (versión no estacionaria), Sistema de Colonias de Hormigas.

1. INTRODUCCIÓN

La técnica de Optimización de Colonias de Hormigas (Dorigo et al. [10]) es una metaheurística bio-inspirada aplicable a problemas de optimización combinatorios duros en ambientes estacionarios y no estacionarios. Los algoritmos derivados de la misma son esencialmente algoritmos aproximados, es decir, proporcionan soluciones de alta calidad (aunque no necesariamente la óptima) en un tiempo computacional breve. Los algoritmos ACO se inspiran en el comportamiento colectivo y cooperativo que rige a las hormigas de diversas especies para encontrar los caminos más cortos entre las fuentes de comida y el hormiguero. Algunas especies de hormigas se comunican a través del depósito de una sustancia denominada feromona, lo que da lugar a un esquema de comunicación indirecta. Los algoritmos ACO involucran una colonia de hormigas artificiales, agentes computacionales simples que trabajan de manera cooperativa y se comunican mediante rastros de feromona artificiales. Las hormigas artificiales son procedimientos de construcción estocásticos que probabilísticamente construyen una solución agregando iterativamente componentes del problema a las soluciones parciales del problema teniendo en cuenta: (i) información heurística de la instancia del problema a ser resuelto, si está disponible, (ii) el rastro de feromona, el cual cambia dinámicamente en tiempo de ejecución para reflejar la experiencia de búsqueda adquirida.

El *Problema de los ascensores múltiples* (MEP), es un problema de optimización combinatorio no estacionario, que consiste en encontrar una secuencia de movimientos que deben realizar los ascensores de un edificio de manera tal de minimizar el tiempo medio de espera de los pasajeros. La llegada de un nuevo pasajero a la cola de un ascensor, la rotura de un ascensor, etc, son eventos que provocan cambios de estado en el problema, haciéndolo un problema no estacionario ya que se modifica la instancia del problema para la cual se intenta encontrar una solución. El MEP es de gran interés debido a su aplicabilidad en el mundo real, tiene muchos puntos en común con los procesos físicos industriales, en ellos, existen una serie de sensores que reflejan sus cambios de estado. Estos cambios obligan a la búsqueda continua de un óptimo móvil, y en consecuencia se llevan a cabo una serie de acciones.

En este trabajo se presenta un modelo ACO para una versión no estacionaria del *Problema del Ascensor Único* (SEP) orientado a un algoritmo ACS. En SEP, se tiene en cuenta un único ascensor y para éste se, intenta encontrar la secuencia de visitas que debe realizar de manera tal de minimizar el tiempo medio de espera de los pasajeros, esperando en alguna de las colas de las plantas del edificio. Se dice un problema no estacionario debido a que no se consideran los cambios que se pueden producir eventualmente en la instancia del problema. Para la definición del modelo ACO, se adoptan características del SEP utilizadas en Torres [11] (una configuración estática) para el diseño de un simulador del funcionamiento de un ascensor, en donde se aplica un enfoque evolutivo para la obtención de la secuencia de visitas.

2. EL PROBLEMA DE LOS ASCENSORES MÚLTIPLES

La formulación del problema general tiene su fundamento en los llamados Sistemas de Ascensores Múltiples. En términos generales, un sistema idealizado de ascensores múltiples consiste de m ascensores en un edificio de n plantas en donde los usuarios, las personas que necesitan trasladarse de una planta a otra del edificio, generan llamadas a los ascensores. Las llamadas pueden ser originadas desde cualquier planta o desde el interior de los ascensores: subir a una planta, bajar a una planta, detener el ascensor, abrir la puerta del ascensor, etc. Ante este tipo de llamadas, cada ascensor debe realizar un recorrido eficiente de las plantas, siguiendo una secuencia de visitas de longitud k de manera tal que se minimice el tiempo medio de espera de los usuarios, el tiempo medio de tránsito de

los usuarios o la cantidad de usuarios dentro de un ascensor.

Existe cierto dinamismo subyacente al problema. Después de un período de tiempo, dentro del horario de actividad de los ascensores, puede ocurrir que varíe: la cantidad de llamadas de los usuarios, la cantidad de ascensores en funcionamiento, las prioridades de las plantas a visitar, etc. Esto implica un cambio en la decisión sobre qué secuencia de visitas debería seguir cada ascensor.

Para el MEP se han realizado investigaciones focalizadas en la construcción de los controladores en sí, otras dirigidas a la optimización de parámetros del controlador de un sistema de ascensores, como así también investigaciones dedicadas a los algoritmos de reoptimización. Por ejemplo, para la construcción del controlador en sí: Ho et al. [5] plantea el uso de Redes de Petri combinadas con Redes Neuronales para aprender la mejor política de planificación, Kojima et al. [14] aplican Computación-ADN para minimizar los tiempos de espera. Trabajos más recientes vinculados al uso del enfoque evolutivo, incluyen a Markon et al. [2] en el cual se aplica Programación Red-Genética (Genetic Network Programming) para evolucionar un controlador de un sistema de ascensores múltiples. Otros enfoques aplican controladores difusos, sistemas expertos, redes neuronales y combinación de los mismos, los cuales muestran otros enfoques alternativos del problema a abordar [1, 4, 5, 6, 7, 8, 9]. Para la optimización de parámetros de un controlador del sistema de ascensores, se asume una unidad de control que usa una serie de parámetros para tomar las decisiones respecto a la asignación de ascensores cada vez que son llamados. El objetivo aquí es determinar los valores para dicho conjunto de parámetros que optimicen los criterios establecidos. Por ejemplo, minimización de tiempos de espera (waiting times), reducción del factor de aglomeración (crowding factor), minimización del tiempo de traslado (riding time), etc. Los trabajos de Fujino et al. [13, 3] muestran la aplicación de algoritmos genéticos (AGs) para ajustar un conjunto de parámetros usados en el control de ascensores múltiples. Para los algoritmos de reoptimización, el trabajo de Friese et al. [12] muestra un conjunto de algoritmos de re-optimización para un sistema restringido de ascensores. En el mismo se presentan y prueban dos modelos - estacionario y no estacionario - y una serie de políticas de asignación y planificación.

2.1. El Problema del Ascensor Único (versión no estacionaria)

El SEP es una subclase del problema de ascensores múltiples. Se tiene un edificio de n plantas con un único ascensor, $m = 1$, donde el objetivo es minimizar el tiempo medio de espera de los usuarios del ascensor. El ascensor debe ir siguiendo la secuencia de visitas de plantas especificada atendiendo a su paso el máximo número de peticiones procurando el mínimo tiempo de espera para los usuarios. En este trabajo se adoptan características del modelo de único ascensor establecido por Torres [11]: (i) la secuencia de visitas está compuesta de números enteros representando una de las posibles plantas del edificio. Cada una de estas plantas es una planta destino del ascensor, (ii) el ascensor se detiene en cada planta que haya algún pasajero cuya planta destino sea igual al destino del ascensor, teniendo siempre en cuenta la capacidad máxima del ascensor, (iii) se respeta una configuración estática para la inicialización de las llamadas del ascensor. Se considera una única serie temporal de llamadas al ascensor para producir una secuencia de visitas del ascensor a las plantas, (iv) la función para la evaluación de la calidad de las secuencias de visitas.

a. Formulación del problema

En el SEP, el ascensor debe realizar un recorrido del edificio siguiendo una secuencia de visitas $s = \langle p_1, p_2, \dots, p_k \rangle$ de longitud k , que puede ser visualizada como una lista de número enteros, representando alguna de las n plantas del edificio. No pueden haber números contiguos iguales, ya

que un ascensor no puede tener como próximo movimiento la planta en donde se encuentra en ese momento detenido. El primer número de la secuencia, p_1 , corresponde a la planta de partida del ascensor. El ascensor partiendo de la primer planta de la secuencia, considerará como planta destino la siguiente indicada en la secuencia y parará en todas las plantas intermedias para recoger a los usuarios que vayan a dicha planta destino, teniendo siempre en cuenta la capacidad máxima del ascensor ($C_{\text{máx}}$).

Por ejemplo, la Figura 1 muestra una secuencia posible de visitas de longitud $k = 8$. El ascensor comienza su recorrido desde la planta 2, luego realiza un primer movimiento siendo el destino la planta 3, continúa hacia la planta 5, luego, la planta 6 es elegida como tercer movimiento, y así sucesivamente hasta completar el recorrido haciendo el movimiento $k - 1 = 7$ hacia la planta 10.

1	2	3	4	5	6	7	8
2	3	5	6	4	3	6	10

Figura 1: Una secuencia de visitas de longitud $k = 8$, es decir, $k - 1 = 7$ movimientos.

Para elegir la secuencia de visitas que minimice el tiempo medio de espera de los usuarios, se tiene en cuenta la siguiente información referente al problema: (i) μ_i^t , el tiempo medio de espera total de las personas en la planta i ; (ii) μ_{ij} , el tiempo medio de espera de una persona j en la planta i y (iii) β_i , el número de personas que han esperado en la planta i .

Luego, la formulación del problema es equivalente a encontrar la secuencia que minimice el valor de la siguiente función:

$$f = \frac{1}{n} \sum_{i=1}^n \mu_i^t \quad \text{Donde:} \quad \mu_i^t = \frac{1}{\beta_i} \sum_{j=1}^{\beta_i} \mu_{ij} \quad (1)$$

b. Definición una instancia del SEP como un problema combinatorio

En términos de problemas combinatorios, el SEP se puede definir como:

$$\pi = (S, f, \Omega)$$

donde S es el conjunto de soluciones candidatas, posibles secuencias de visitas de longitud k ; f es la función objetivo, la cual, asigna un valor de función objetivo $f(s)$ a cada solución candidata $s \in S$ y finalmente, Ω es un conjunto de restricciones. Para el SEP la única restricción consiste en que no se permite en posiciones contiguas de la secuencia un mismo número (o planta). La capacidad máxima del ascensor ($C_{\text{máx}}$) no forma parte del conjunto de restricciones ya que la secuencia establecida como solución sigue siendo válida en caso de alcanzar dicha capacidad máxima.

Las soluciones correspondientes al conjunto $\tilde{S} \subseteq S$ de soluciones candidatas que satisfacen las restricciones Ω son llamadas *soluciones factibles*. El propósito es encontrar una *solución* $s^* \in \tilde{S}$ *factible óptima global*.

2.2. Modelo ACO para el SEP

En un enfoque ACO las hormigas artificiales construyen una solución en forma incremental, formando soluciones parciales a través de la inserción de componentes, de a una por vez. Por lo tanto, un problema se puede resolver con un algoritmo ACO (Dorigo et al. [10]) si puede ser transformado a una representación conveniente para que las hormigas puedan realizar este proceso constructivo, para formar una solución.

El problema combinatorio del ascensor único $\pi = (S, f, \Omega)$, es transformado sobre un problema que puede ser caracterizado por los siguientes items:

- Un conjunto $P = \{p_1, p_2, \dots, p_n\}$ de componentes, las plantas de un edificio, donde n es el número de componentes;
- Los estados del problema están definidos en términos de secuencias $x = \langle a, p_j, \dots, p_h, \dots, p_q \rangle$ de longitud finita $|x|$ sobre los elementos de P . El conjunto de todos los posibles estados es denotado por χ . La máxima longitud de una secuencia está limitada por una constante positiva $k < \infty$, donde k es la longitud de la secuencia que representa una solución del problema;
- El conjunto de soluciones candidatas S es un subconjunto de χ ;
- Un conjunto de estados factibles $\tilde{\chi}$, con $\tilde{\chi} \subseteq \chi$, el conjunto de las secuencias x satisfaciendo las restricciones Ω . Para el problema del ascensor único, son las secuencias de longitud k donde posiciones contiguas tienen un número de planta diferente;
- Un conjunto no vacío de soluciones óptimas $S^* \subseteq S$, las secuencias que evaluadas con la función objetivo f alcanzan un valor mínimo.
- Un costo $g(s)$ está asociado con cada solución candidata $s \in S$. En el caso del SEP $g(s) \equiv f(s), \forall s \in \tilde{S}$.

Dando esta formulación para el SEP, las hormigas artificiales construyen una solución recorriendo aleatoriamente un camino sobre un *grafo de construcción* G definido a continuación.

a. Definición del grafo de construcción

Para el problema del ascensor único, las hormigas artificiales son las encargadas de construir la secuencia de visitas, realizando un camino aleatorio sobre un grafo de construcción $G = (P, L)$ donde P es el conjunto de plantas (definido en la sección 2.2) y L es el conjunto de arcos que conectan a las componentes de P . Si existiera una conexión directa entre cada par de nodos del grafo (como el grafo del problema TSP), se estaría representando de esta manera la posibilidad que tiene el ascensor de ir de una planta a cualquier otra planta del edificio, es decir, se estaría representando la posibilidad que tiene una hormiga de pasar por cualquier otro nodo del grafo. Entonces, considerando la información del rastro entre las plantas y que una secuencia de visitas no es una permutación, una hormiga podría en su recorrido, pasar más de una vez por el mismo nodo, teniendo similares posibilidades de seguir el mismo recorrido ya que no puede diferenciar en qué etapa del recorrido se encuentra. Como una alternativa a este grafo se propone el *grafo de construcción parcialmente conectado*.

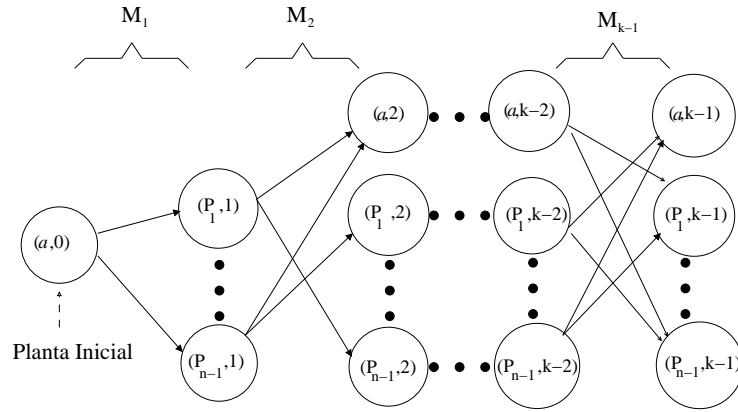


Figura 2: Grafo parcialmente conectado.

El grafo de construcción parcialmente conectado

Es un grafo dirigido con $n \times (k - 1)$ nodos rotulados con algún par: (*número de planta, número de movimiento*), indicando que el ascensor visitará la planta "*número de planta*" luego de realizar el movimiento "*número de movimiento*", y un conjunto de arcos representando los posibles movimientos entre las plantas.

La Figura 2, muestra un grafo parcialmente conectado donde:

- $a \in \{1 \dots n\}$ es la planta inicial, que se alcanza en 0 movimientos,
- $P_k \in \{1, \dots, n - 1\}$ es cualquier planta distinta a la planta a ,
- cada hilera de nodos corresponde a las plantas que puede visitar el ascensor en un mismo movimiento (M_1, M_2, \dots, M_{k-1}). Entre dos movimientos consecutivos no existe una conexión directa a los pares que representan la misma planta (que tienen igual *número de planta*).

Si la secuencia de plantas de la solución tiene longitud k , luego la cantidad de movimientos es $k - 1$. Como a no puede ser elegida para el movimiento M_1 , solo $n - 1$ nodos pueden ser elegidos para él, debido a que $(a, 1)$ no es alcanzable desde $(a, 0)$.

Por ejemplo, si $P = \{1, 2, 3, 4, 5\}$, $k = 4$, $n = 5$, $a = 3$, $P_1 = 1$, $P_2 = 2$, $P_3 = 4$, $P_4 = 5$, luego las posibles secuencias de longitud k se corresponden a cada uno de los *número de planta* de los nodos que conectan los arcos formando un camino (Figura 3).

Antes de continuar con las definiciones que competen a una metaheurística ACO, cabe destacar, que para cada una de ellas se resaltarán las características pertinentes a los algoritmos ACS, ya que, el propósito de la definición del modelo que se presenta, es que sea aplicado a un algoritmo de este tipo.

b. La estructura de feromonas τ

La idea básica respecto a la matriz de feromonas τ descrita en Dorigo et al. [10], consiste de una estructura que contiene valores que hacen referencia a la cantidad de feromona acumulada que depositan las hormigas durante su recorrido. Para el SEP, según el grafo descrito anteriormente, τ es un conjunto de $k - 1$ estructuras $\tau_{M_1} \dots \tau_{M_{k-1}}$, correspondiente a cada movimiento realizado por el ascensor al seguir la secuencia. Para el primer movimiento la dimensión de τ_{M_1} es $n - 1$, ya que se

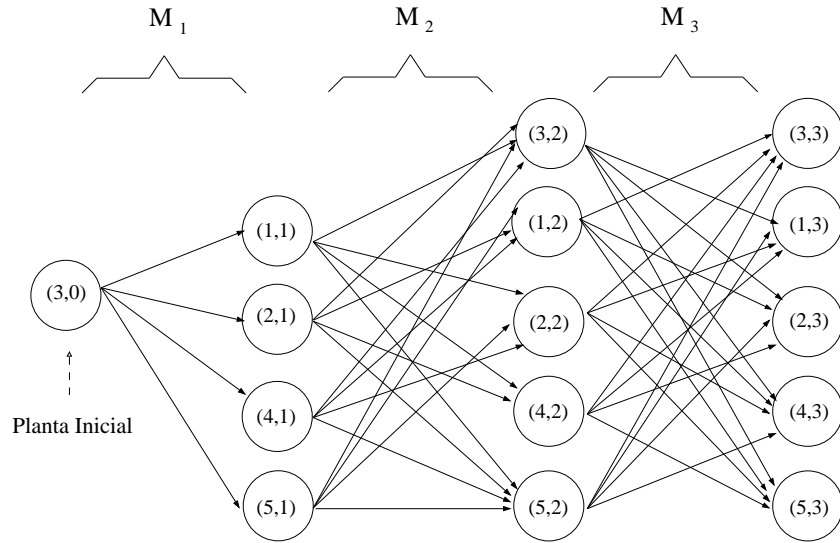


Figura 3: Grafo parcialmente conectado para $P = \{1, 2, 3, 4, 5\}$, $k = 4$ y $a = 3$.

debe considerar únicamente los posibles movimientos desde la planta a (la planta inicial) a cualquier otra planta, para el resto de los movimientos, el tamaño de cada τ_{M_j} es $n \times n$.

c. Modificación de la estructura de feromona τ

Un algoritmo ACS es una versión mejorada de un *Sistema de Hormigas* (AS) el cual incrementa la explotación de la información recolectada por la colonia de hormigas para guiar la exploración del espacio de búsqueda. Existen dos mecanismos para este propósito. La primera es usar una estrategia elitista para modificar el rastro de feromona y la segunda, modifica el mecanismo original para la selección de la próxima componente de la solución (explicado en el siguiente punto). En cuanto a la actualización del rastro de feromona, se aplican dos reglas: *modificación local* y *modificación global* (ecuaciones (2) y (3)).

Regla de modificación local:

cada vez que una hormiga h selecciona una planta j para el movimiento x del ascensor, detenido en la planta i , se modifican los niveles de feromona de la estructura $\tau_{M_x}[i, j]$ sumando un valor constante τ_0 a su valor actual.

Regla de modificación global:

luego de que cada hormiga construye una secuencia, se elige la secuencia con la cual se obtuvo el mínimo valor para la función (1) (construida por la hormiga h^*).

La modificación de cada τ_{M_x} , se realiza teniendo en cuenta el valor de $\Delta\tau_{M_x}[i, j]^{h^*}$, definida de la siguiente manera:

$$\Delta\tau_{M_x}[i, j]^{h^*} = \begin{cases} \frac{TME}{TT} & \text{si la hormiga } h^* \\ & \text{recorre el arco } ((i, x-1), (j, x)) \\ 0 & \text{en otro caso} \end{cases} \quad (2)$$

Luego la matriz τ_{M_x} se actualiza de la siguiente manera:

$$\tau_{M_x}[i, j] = \rho \times \tau_{M_x}[i, j] + \Delta\tau_{M_x}[i, j]^{h^*} \quad (3)$$

donde,

- $\rho \in [0, 1]$ es el parámetro que regula la reducción de $\tau_{M_x}[i, j]$ (la evaporación del rastro de feromona),
- $\Delta\tau_{M_x}[i, j]^{h^*}$ cambio del nivel del rastro causado por la hormiga h^* ,
- TME es el tiempo medio de espera de las personas esperando en las plantas que conforman la secuencia encontrada por una hormiga h^* ,
- TT es el tiempo total del funcionamiento del ascensor.

d. La regla para la selección de una planta

Respecto a la elección de la próxima planta para la construcción de una solución de la hormiga h , se utiliza una *regla proporcional pseudo-random*. Si la hormiga h está en la planta i , la probabilidad de ir a la planta $j \in \tilde{\chi}$ en el movimiento x , es:

Con probabilidad q_0 ,

$$j = \arg \max_{j \in \tilde{\chi}} \{ \tau_{M_x}[i, j][\eta_{ij}]^\beta \} \quad (4)$$

en otro caso la próxima componente es elegida acorde a la ecuación:

$$p_{ijx}^h = \frac{\tau_{M_x}[i, j][\eta_{ij}]^\beta}{\sum_{l \in \mathcal{N}_i^h} \tau_{M_x}[i, l][\eta_{il}]^\beta} \quad (5)$$

donde $\mathcal{N}_i^h = P - \{i\}$ representa el conjunto de plantas distintas a i , η_{ij} es la información heurística y β es el parámetro que determina la influencia relativa de la información heurística. En este modelo se propone la siguiente información heurística de la planta j : (a) la distancia entre la planta i y la planta j , (b) la longitud de cola, (c) el tiempo medio de espera y (d) la cantidad de pasajeros que están dentro del ascensor y descienden en esta planta. Luego,

$$\eta_{ij} = \alpha_1 \left(\frac{1}{d_{ij}} \right) + \alpha_2 Q_j + \alpha_3 AQ_j + \alpha_4 R_j \quad (6)$$

donde

(a) d_{ij} es la distancia de la planta actual i a la planta j

(b) $Q_j = \frac{\text{Cantidad de personas esperando en el planta } j}{\text{Cantidad total de personas esperando en el edificio}}$

(c) $AQ_j = \frac{\text{Tiempo medio de espera en la planta } j}{\text{Tiempo medio de espera total}}$

(d) $R_j = \frac{\text{Cantidad de llamadas en el planta } j}{\text{Cantidad total de llamadas en el edificio}}$

Según la ecuación (6), la influencia relativa de cada uno de los términos está determinada por los pesos $\alpha_1 \dots \alpha_4$. Por ejemplo, si se le da igual importancia a todos los valores de dicha ecuación (i.e., $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4$), tendrá mayor probabilidad de ser elegida aquella planta que está a menor distancia de i , con una mayor cantidad de personas esperando, mayor tiempo medio de espera y en la cual descende un mayor número de pasajeros.

e. Algoritmo ACO para el SEP

En este punto se presenta el algoritmo *Ascensores-ACS* (Algoritmo 1), un algoritmo ACS basado en el modelo ACO propuesto anteriormente. Tal como se mencionó anteriormente, el ACS incrementa la explotación de la información recolectada por la colonia de hormigas para guiar la exploración del espacio de búsqueda. Los mecanismos utilizados son: la utilización de una estrategia elitista para modificar el rastro de feromona y la aplicación de la *regla proporcional pseudo-random* descrita en la ecuaciones (4) y (5). Cuando $q_0 = 1$, el ACS es altamente greedy, así las plantas que se favorecen son las que poseen una mejor combinación de niveles altos de feromona y valores heurísticos. Para el otro caso extremo en el que $q_0 = 0$ el ACS adopta un comportamiento similar a AS. Por lo tanto, el valor dado a q_0 determina el balance entre una selección voraz y probabilística de la próxima componente. Adicionalmente, se aplican dos reglas para la modificación de los niveles de feromona: *regla de modificación local* y la *regla de modificación global* descritas en el punto c.

```

proc Ascensores – ACS
  begin
    inicializar();
    for (t=1 ... T) do
      for (h=1 ... m) do
        secuencia[1] = planta_inicial;
        for (i=2 ... k) do
          secuencia[i] = seleccionarPlanta();
          moverAscensor();
          actualizarTiempos();
          modificacionLocal();
        od
        if ( $f(\textit{secuencia}) < f(\textit{secuenciaOptima})$ )
          secuenciaOptima = secuencia;
        fi
      od
      modificacionGlobal();
    end
  end

```

Algoritmo 1: ACS para el SEP

El algoritmo *Ascensores-ACS* en cada iteración h genera una secuencia de visitas denominada *secuencia*. Cada componente de esta secuencia es obtenida a través de la función *seleccionarPlanta()*, excepto la primer planta (*secuencia*[1]) que corresponde a la planta en donde se encuentra detenido el

ascensor antes de iniciar su recorrido. Luego de seleccionar una planta se debe simular el recorrido del ascensor hasta la última planta seleccionada (considerando los descensos y ascensos de los pasajeros según la formulación del problema). Esta acción es realizada usando la función *moverAscensor()* y como consecuencia de esto se producen cambios respecto al estado de las colas, la cantidad de pasajeros puede variar y por lo tanto pueden cambiar los tiempos promedios, por lo que es necesario realizar las actualizaciones pertinentes. Para ello, se aplica la función *actualizarTiempos()*.

Durante todo el proceso ACO se va actualizando la *secuenciaOptima* que es la estructura encargada de mantener la secuencia de visitas con menor valor de f , i.e., menor tiempo promedio de espera.

Las principales componentes del algoritmo *Ascensores-ACS* se describen a continuación:

- *inicializar()*
Se inicializan las estructuras propias del algoritmo ACO, se realizan las lecturas de los archivos de entradas conteniendo aquellos parámetros de entradas del algoritmo ACO y la instancia del problema a considerar;
- *seleccionarPlanta()*
Es una función que retorna el número de planta en el que se detendrá el ascensor en el próximo movimiento. Aquí se aplican las ecuaciones (4) y (5);
- *moverAscensor()*
Es una función que simula el recorrido del ascensor hasta la última planta seleccionada (considerando los descensos y ascensos de los pasajeros).
- *actualizarTiempos()*
Actualiza los tiempos de espera de las personas esperando en el edificio considerando el movimiento del ascensor a la planta recientemente elegida;
- *modificacionLocal()*
Actualiza el nivel de feromona según la nueva planta seleccionada, se aplica la *regla de modificación local*;
- *modificacionGlobal()*
Actualiza el nivel de feromona según la mejor secuencia obtenida en la iteración (*secuenciaOptima*) se aplica la *regla de modificación global*, ecuaciones (2) y (3);
- $f()$
Es la función que calcula el tiempo promedio de espera, ecuación (1).

3. EXPERIMENTOS PRELIMINARES

Esta sección presenta un estudio preliminar del algoritmo ACO propuesto, sobre dos instancias de SEP. Dicho estudio consistió en mostrar los posibles valores de los distintos parámetros que sean más adecuados para el desempeño del algoritmo, i.e., aquellos que permitan obtener una secuencia de visitas de tamaño k que minimice el tiempo medio de espera μ de los pasajeros. Para ello, se consideró una configuración estática del problema, es decir, durante toda la ejecución del algoritmo se mantuvo la misma secuencia temporal de llamadas al ascensor, las cuales se guardan en un archivo para luego ser leídas por el algoritmo ACO. Dicha secuencia de llamadas es representada por una lista $L = \langle (t^1, p_o^1, p_d^1), \dots, (t^l, p_o^l, p_d^l) \rangle$ de longitud l en donde: t^i es el tiempo de ocurrencia de la llamada y se cumple $\forall i = 1 \dots l - 1, t^i < t^{i+1}$, p_o^i es la planta de donde se origina la llamada

y p_d^i es la planta destino. El tiempo entre ocurrencias de llamadas tiene una distribución uniforme y se generaron dos instancias para las cuales se consideraron los intervalos $[10, 60]$ y $[10, 300]$ (en segundos) respectivamente. Para ambos casos se consideró una secuencia L de longitud $l = 1000$. El escenario para el ascensor fue establecido como sigue: posición inicial del ascensor $a = 0$, capacidad del ascensor $c = 8$, número de plantas del edificio $n = 10$, longitud de la secuencia de visitas $k = 1000$ y el tiempo que tarda el ascensor en desplazarse de un piso a otro $ta = 2$ segundos. Mientras que para el algoritmo ACO: número de hormigas $m = 10$ y los restantes valores para q_0 , β , τ_0 , ρ , α_1 , α_2 , α_3 y α_4 , se muestran en las Tablas 1 y 2 correspondientes a las instancias 1 y 2 respectivamente. Para cada conjunto de valores de parámetros se realizaron 20 ejecuciones a partir de las cuales se calcula el valor medio μ (en segundos) de las medias de cada ejecución, es decir, $\mu = \frac{1}{20} \sum_{j=1}^{20} \mu_j$.

Se puede observar que para ambas instancias, los experimentos $E11$ y $E12$ presentan los mejores resultados, esto es un indicador que el algoritmo se comporta mejor cuando se aplica la ecuación de selección (4), cuando el algoritmo tiene un comportamiento altamente voraz, y cuando la regla de modificación local produce grandes cambios en los valores de feromona (τ_0). Sin embargo, un estudio en profundidad deberá ser realizado a fin de establecer la calidad del algoritmo propuesto.

E	T	q_0	β	τ_0	ρ	α_1	α_2	α_3	α_4	μ
E1	50	0.3	0.5	0.1	0.2	0.05	0.5	0.1	0.35	384.021
E2	50	0.3	0.5	0.1	0.2	0.05	0.5	0.15	0.3	383.516
E3	50	0.3	0.5	0.1	0.2	0.05	0.15	0.5	0.3	381.656
E4	50	0.3	0.5	0.1	0.2	0.05	0.3	0.5	0.15	381.862
E5	100	0.3	0.5	0.1	0.2	0.05	0.5	0.7	0.15	381.496
E6	100	0.3	0.5	0.1	0.2	0.05	0.3	0.7	0.5	380.973
E7	100	0.3	0.5	0.1	0.2	0.05	0.3	0.5	0.7	381.249
E8	100	0.5	0.5	0.8	0.5	0.05	0.3	0.7	0.5	374.427
E9	150	0.3	0.5	0.1	0.2	0.05	0.5	0.7	0.15	381.496
E10	150	0.3	0.5	0.1	0.2	0.05	0.3	0.7	0.5	380.973
E11	100	0.8	0.5	0.8	0.5	0.05	0.3	0.7	0.5	364.396
E12	100	0.8	0.5	0.8	0.5	0.05	0.7	0.5	0.3	364.477

Tabla 1: Tiempos medios de espera en cola para la instancia 1.

E	T	q_0	β	τ_0	ρ	α_1	α_2	α_3	α_4	μ
E1	50	0.3	0.5	0.1	0.2	0.05	0.5	0.1	0.35	1537.059
E2	50	0.3	0.5	0.1	0.2	0.05	0.5	0.15	0.3	1534.618
E3	50	0.3	0.5	0.1	0.2	0.05	0.15	0.5	0.3	1526.080
E4	50	0.3	0.5	0.1	0.2	0.05	0.3	0.5	0.15	1528.968
E5	100	0.3	0.5	0.1	0.2	0.05	0.5	0.7	0.15	1527.287
E6	100	0.3	0.5	0.1	0.2	0.05	0.3	0.7	0.5	1523.111
E7	100	0.3	0.5	0.1	0.2	0.05	0.3	0.5	0.7	1522.732
E8	100	0.5	0.5	0.8	0.5	0.05	0.3	0.7	0.5	1493.599
E9	150	0.3	0.5	0.1	0.2	0.05	0.5	0.7	0.15	1527.287
E10	150	0.3	0.5	0.1	0.2	0.05	0.3	0.7	0.5	1523.111
E11	100	0.8	0.5	0.8	0.5	0.05	0.3	0.7	0.5	1462.159
E12	100	0.8	0.5	0.8	0.5	0.05	0.7	0.5	0.3	1461.955

Tabla 2: Tiempos medios de espera en cola para la instancia 2.

4. CONCLUSIONES

En este trabajo se presentó una propuesta de un modelo ACO para una versión no estacionaria del *Problema del Ascensor único*. La definición de este modelo se orientó hacia el ACS, un tipo avanzado de algoritmo ACO. Esto fué posible debido a que se pudieron: definir la estructura de feromonas (y las

funciones para su mantenimiento), adaptar las reglas de selección pseudo-aleatoria que rigen al ACS para el SEP y representarlo a través de un *grafo de construcción parcialmente conectado*, de manera tal que las hormigas artificiales puedan encontrar una solución, realizando el proceso constructivo que las caracteriza sobre este grafo. Si bien el tamaño del *grafo de construcción parcialmente conectado* crece de manera importante cuando n (el número de plantas) y k (la longitud de la secuencia de visitas) aumentan, se considera un buen punto de partida para el estudio del SEP y sus posibles extensiones, como lo son: la versión dinámica del problema del Ascensor único y el problema de los Ascensores Múltiples. Por último, un estudio experimental extensivo será necesario para validar el algoritmo propuesto para SEP.

REFERENCIAS

- [1] Zhu Dewen, Jiang Li, Zhou Yuwen, Shan Guanghui, and He Kai. Modern elevator group supervisory control systems and neural networks technique. In *1997 IEEE International Conference on Intelligent Processing Systems, ICIPS '97.*, pages 528–532, 28-31 Oct. 1997.
- [2] T. Eguchi, K. Hirasawa, Jinglu Hu, and S. Markon. Elevator group supervisory control systems using genetic network programming. In *Congress on Evolutionary Computation, 2004. CEC2004*, volume 2, pages 1661–1667, 19-23, June 2004.
- [3] A. Fujino, T. Tobita, K. Segawa, K. Yoneda, and A. Togawa. An elevator group control system with floor-attribute control method and system optimization using genetic algorithms. In *IEEE Transactions on Industrial Electronics*, volume 44, pages 546–552, Aug. 1997.
- [4] R. Gudwin, F. Gomide, and M. Andrade Netto. A fuzzy elevator group controller with linear context adaptation. In *IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence*, pages 481–486, 4-9 May 1998.
- [5] Ming Ho and B. Robertson. Elevator group supervisory control using fuzzy logic. volume 2, pages 825–828, 25-28 Sept.
- [6] N. Imasaki, S. Kubo, S. Nakai, T. Yoshitsugu, Jun-Ichi Kiji, and T. Endo. Elevator group control system tuned by a fuzzy neural network applied method. In *Proceedings of 1995 IEEE International Conference on Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium*, volume 4, pages 1735–1740, 20-24 March 1995.
- [7] T. Ishikawa, A. Miyauchi, and M. Kaneko. Supervisory control for elevator group by using fuzzy expert system which also addresses traveling time. In *Proceedings of IEEE International Conference on Industrial Technology 2000.*, pages 87–94, 19-22 January 2000.
- [8] Chang Bum Kim, K.A. Seong, Hyung Lee-Kwang, and J.O. Kim. Design and implementation of a fuzzy elevator group control system. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 28(3):277–287, May 1998.
- [9] Chang Bum Kim, K.A. Seong, Hyung Lee-Kwang, J.O. Kim, and Yong Bae Lim. A fuzzy approach to elevator group control system. *IEEE Transactions on Systems, Man and Cybernetics*, 25(6):985–990, June 1995.
- [10] Dorigo Marco and Stützle Thomas. *Ant Colony Optimization*. Mit Press, 2004.

- [11] Rafael Torres Márquez. Algoritmos evolutivos distribuidos en entornos dinámicos. Master's thesis, Departamento Lenguajes y Ciencias de la Computación - Universidad de Málaga, 2000.
- [12] J. Rambaud and P. Friese. On line-optimization of a multi-elevator transport system with reoptimization algorithms based on set-partitioning models. Technical report, ZIB Report 05-03, 2003.
- [13] T. Tobita, A. Fujino, K. Segawa, K. Yoneda, and Y. Ichikawa. A parameter tuning method using genetic algorithms for an elevator group control system. In *Proceedings of the 1996 IEEE IECON 22nd International Conference on Industrial Electronics, Control, and Instrumentation*, pages 823–828, 5-10 Aug. 1996.
- [14] J. Watada, S. Kojima, S. Ueda, and O. Ono. DNA computing approach to optimal decision problems. In *International Joint Conference on Neural Networks - IEEE Conference on Fuzzy Systems*, pages 25–29, Budapest, Hungary, July 2004.